REMARKS

Claims 34-49 are pending in the present application. Claims 34-49 have been rejected. Claims 1-33 were previously cancelled. Claim 34 has been amended to further clarify the present invention. No new matter has been added. Accordingly, claims 34-49 are now pending in the present application.

Amended Claim 34

Claim 34 has been amended to better clarify the present invention and present claims for the purposes of allowance or appeal. Claim 34 has been amended without the addition of new matter – and therefore, Applicant believes no further search is required. More specifically, Applicant has incorporated originally filed claims 2-6 within amended claim 34. Applicant has also repositioned language with claim 34 to improve readability of the claim as amended. Applicant asserts no new matter nor any matter requiring a new search has been set forth, and asks Examiner to immediately contact Applicant if Examiner is in disagreement with such.

Double Patenting

Claims 34 and 35 were provisionally rejected under 35 U.S.C. § 101 as claiming the same invention as that of claim 35 and 35 of co-pending Application Number 10/763,135.

Applicant respectfully submits that the claims in the present application as amended and the claims in co-pending U.S. Patent Application No. 10/763,135 are patentably distinct as the present claims in the present application recite elements not found in the newly added claims of co-pending U.S. Patent Application No. 10/763,135, and vice versa. Further, Applicant notes that the claims of 10/763,135 have not issued. Therefore, withdrawal of the provisional double patenting rejection is respectfully requested.

Claim 34, as amended, recites:

34. (Currently Amended) A method for supporting a transaction application workload and a parallel application workload on one server cluster including server nodes at one domain and configured to process the transaction application workload and the parallel application workload, the method comprising:

receiving a request from a client to execute the transaction application workload on the one server cluster~~, the one server cluster including server nodes at one domain~~ the transaction application comprising a plurality of discrete events that are less numerically intensive than the parallel application which comprises a numerically intensive application;

identifying a service level agreement negotiated with the client for the transaction application workload, the service level agreement specifying performance requirements for execution of the transaction application workload on the one server cluster and acceptable performance of execution of the one server cluster in response to the transaction application and the parallel application;

assigning a subset of the server nodes in the one server cluster at the one domain to execute the transaction application workload;

monitoring execution of the transaction application workload on the subset of server nodes assigned to execute the transaction application workload to determine whether the performance requirements for execution of the transaction application workload and the acceptable performance of execution of the one server cluster in response to the transaction application and the parallel application specified in the service level agreement are being met; and

responsive to a determination that the performance requirements for execution of the transaction application workload specified in the service level agreement are not being met, dynamically reassigning one or more of the server nodes in the one server cluster at the one domain assigned to execute the parallel application workload to the execution of the transaction application workload in order to meet the performance requirements for execution of the transaction application workload and the acceptable performance of execution of the one server cluster in response to the transaction application and the parallel application specified in the service level agreement,

wherein a server node assigned to execute the transaction application workload cannot be concurrently assigned to execute the parallel application workload and a server node assigned to execute the parallel application workload cannot be concurrently assigned to execute the transaction application workload.

Claim Rejections - 35 USC § 103

Claims 34-49 were rejected by Examiner under 35 U.S.C. § 103(a) as being unpatentable over Herington (U.S. Publication Number 2005/0102387 A1) in view of Sankaranarayan et al. (hereinafter Sankaranarayan) (U.S. Patent No. 6,799,208 B1). Applicant respectfully disagrees.

I. ***Herington does not disclose a method for supporting a transaction application workload and a parallel application workload.***

Examiner has asserted that: "Regarding claim 34, Herington teaches as follows: a method for supporting a transaction application (interpreted as one application 255 in figure 2) workload and a parallel application (interpreted as the other application 260 in figure 2) workload…

Applicant responds that Herington discloses only clients issuing transactions to applications via network and does not disclose, teach or otherwise suggest transaction application workloads and parallel application workloads. The present invention explicitly sets forth differences as between the application types and the operations of which are undertaken by the differing applications, whereas

Herington fails to do so for Herington never envisioned such and was never intended to operate with regard for such.

For instance, in the instant application, at [0008], Applicant states: “Transaction applications are comprised of tasks that are small discrete events such as, for example, stock trading transactions. Parallel tasks are numerically intensive tasks such as, for example, a stock portfolio optimization.” Operationally, these tasks have specific demands and require resources different from one another. Herington fails to account for differing needs of differing application workload types as Herington is at best an attempt at generic load balancing for a specific configuration. Applicant believes Herington fails to disclose such as Herington is not instructive in a load balancing method to accommodate such. Further, both Examiner’s own basis for rejection and Herington’s teachings render clear that the rejection is unfortunately premised on mere supposition and speculation.

Examiner states: “supporting a transaction application (***interpreted as one application 255 in figure 2***) workload and a parallel application (***interpreted as the other application 260 in figure 2***) workload…” Applicant requests specificity of an express and explicit teaching for such interpretation as even Herrington teaches contrary to such supposition through the Specification where application 255 and application 260 are used interchangeably and synonymously throughout.

Applicant notes:, under MPEP § 2112:

“To establish inherency, the extrinsic evidence ‘must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill. Inherency, however, may not be established by probabilities or possibilities. The mere fact that a certain thing may result from a given set of circumstances is not sufficient.' " In re Robertson, 169 F.3d 743, 745, 49 USPQ2d 1949, 1950-51 (Fed. Cir. 1999) (citations omitted).

(MPEP § 2112, 8th ed., Sept. 2007 rev.).

Persons of ordinary skill in the art readily understand that a transaction application workload is not the same as a parallel application workload, and that the demands and performance of each is also different. Therefore, it is <u>NOT</u> inherent that the Herington application(s) may be "interpreted" as suggested by the Examiner, nor by using hindsight of the present invention.

Further Herington expressly states contrary to such supposition at:

1. [0015] Clients 202, 204, 206 may issue transactions ***to applications 255, 260*** via a network 210.

2. [0016] Additionally, …one or more computer resources to applications 245 *[255],* 250.

3. [0017] As the load increases for one of the applications 255, 260, workload balancers 262, 264, 266, 268 may adjust the resources on the nodes to allocate more resources to the application that is experiencing the increased load and fewer resources to the other application.

4. [0018] As the workload managers 262, 264, 266, 268 allocate more resources to an application 255, 260, the load balancers 222, 232 will correspondingly recognize a performance improvement in the node and route more transactions to the node; thus relieving traffic on the other nodes of the cluster.

5. [0027] One or more transactions for application 255 are routed 605 to node 242. Node 242 is a member of both clusters 220, 230 and is executing both applications 255 and 260.

Accordingly, one or more transactions for application 260 may also be routed 610 to node 242.

6. [0028] ***As described previously, one or more computer resources are dynamically allocated 620 to the <u>application</u> 255, 260*** based on performance goals 430 for one or both of the applications and the application performance information 410. Resources may be dynamically allocated 620 by resizing one or more partitions (e.g., virtual partitions, process resource management groupings, etc.) on the node 242. Other methods may also be used to dynamically allocate resources.

Clearly, Herington does not disclose a method for supporting a transaction application workload and a parallel application workload as in the present invention, and Herington is only instructive in that application 255 and application 260 are similar if not identical in operation, contrary to the present invention.

II. ***Herington does not disclose a method for supporting a transaction application workload and a parallel application workload <u>on one server cluster.</u>***

Examiner has asserted that: "Regarding claim 34, Herington teaches as follows: a method for supporting a transaction application (interpreted as one application 255 in figure 2) workload and a parallel application (interpreted as the other application 260 in figure 2) workload **<u>on one server cluster</u>** (220 and 233 in figure 2)(see, e.g., page 1, paragraph [0014].)

Applicant has restated paragraph [0014] of Herington below for convenience:

[0014] An exemplary system for dynamically managing workloads in a cluster is illustrated in FIG. 2. **<u>The system includes two clusters 220, 230</u>**. The clusters each comprise a plurality of nodes 240, 242, 244, 246, 248, 250 (e.g., server computers). **<u>The nodes 240, 242, 244, 246, 248 of the first cluster 220 are each running a first application 255.</u>** **<u>Similarly, the nodes 242, 244,</u>**

> **<u>246, 248, 250 of the second cluster 230 are each running a second application 260</u>**. Each node 242, 244, 246, 248 of cluster 220 appears to clients 202, 204, 206 as a single system. Similarly, each node 242, 244, 246, 248, 250 of cluster 230 appears to the clients 202, 204, 206 as a second single system.

Contrary to Examiner's assertion, but using Examiner's cited portion, Herington clearly and expressly states that there are two clusters. Additionally, Herington explicitly states that applications 255 and 260 are on separate (different) clusters, and hence Applicant asserts such are also at different domains. Hence, Applicant requests Examiner take notice that not only are there two clusters of interest, but that the similar if not identical applications (255,260) of Herington being involved in the load balancing of Herington, are not situated on one cluster per se nor at one domain (ref. Figures of Herington).

Clearly, Herington does not disclose a method for supporting a transaction application workload and a parallel application workload on one server cluster as in the present invention.

III. ***Herington does not disclose receiving a request from a client to execute the transaction application workload on the one server cluster, the one server cluster including server nodes at one domain.***

Incorporating the reasons above, Applicant respectfully disagrees with Examiner's position rejecting claim 34 in part that:

> "the method comprising: receiving a request from a client (clients 202-206 in figure 2) to execute the transaction application workload on the one server cluster (cluster 220 or 230 in figure 2)(clients issue transactions to application via a network to communicate with clusters, see, e.g., page 1, paragraph [0015]), the one server cluster including server nodes (nodes 240-250 in figure 2) at one domain (the clusters each comprise a plurality of nodes, see, e.g., page 1, paragraph [0014]);

Applicant renews their position that the applications are similar if not identical and that Herington is not perform the method of the present invention, nor does so on one server cluster.

Further Herington does not "execute the transaction application workload on the one server cluster" as suggested by Examiner. Examiner has asserted that such is performed by Herington as "clients issue transactions to application via a network to communicate with clusters, see, e.g., page 1, paragraph [0015])." Applicant has already set forth remarks concerning [0015] and renews them here. Further, however, Herington is issuing transaction instructions via the network to applications – and not executing a transaction application workload on one server cluster at one domain – nor a parallel application workload on the one server cluster at one domain.

Accordingly, Applicant believes Herington does not disclose receiving a request from a client to execute the transaction application workload on the one server cluster, the one server cluster including server nodes at one domain.

IV. ***Herington does not disclose identifying a service level agreement negotiated with the client for the transaction application workload, the service level agreement specifying performance requirements for execution of the transaction application workload on the one server cluster.***

Incorporating the reasons above, Applicant respectfully disagrees with Examiner's position rejecting claim 34 in part that:

"identifying a service level agreement negotiated with the client for the transaction application workload, the service level agreement (performance goals, see, e.g., page 2, paragraph [0022]) specifying performance requirements for execution of the transaction application workload on the one server cluster (the incorporated Romero et al. (hereinafter Romero)(U.S. Pub. No. 2002/0069279 A1) teaches an apparatus and method for routing a transaction to a server based on a requested level of service associated with the transaction, see, e.g., abstract);"

Applicant has set forth the included [0022] paragraph reference below:

"FIG. 4 illustrates a first exemplary embodiment of a node that may be one of the intersecting nodes in the system of FIG. 2 or 3. Node 400 includes one or more performance goals 430 for either or both of two applications 402, 404 included on node 400. Performance goals 430 may be a configuration file, information in a database, or other type of stored information that is defined by a user or system administration and may describe the desired level of performance of applications 402, 404. By way of example, performance goals 430 may include response time for the application or particular transactions supported by the application and/or resource utilization goals for the application. Other performance goals are also contemplated. The performance goals 430 and/or applications 402, 404 may each have a relative priority of importance associated with them."

The present invention has set forth at [0033]: "These performance elements comprise the throughput for each application that is supported and, optionally, the response time for the specified throughput." Herington does not contemplate such, as Examiner has pointed to [0022] of Herington to discuss Herington's performance goals for its specific approach.

From Herington and in reviewing Examiner's referenced section, it is apparent that:

1. Herington does not negotiate have a service level agreement negotiated with a client.

2. Herington does not identify a service level agreement negotiated with a client.

3. Herington does not identify a service level agreement negotiated with a client for the transaction workload.

4. Herington does not disclose, teach or otherwise suggest a service level agreement specifying performance requirements for execution of the transaction application workload.

5. Herington does not disclose, teach or otherwise suggest a service level agreement specifying performance requirements for execution of the transaction application workload on the one server cluster.

Further, Herington is express on its limited use of service levels, where:"

[0029] In one embodiment, the method may further include ***monitoring*** service levels provided by each of the nodes 240, 242, 244, 246, 248 in the first cluster 220. Node 242 may

> then be selected based at least in part on its respective service level. Other performance metrics may also be used to select node 242 to satisfy the first transaction.

Even though Romero discusses service levels and a method for routing a transaction to a server based on a requested level of service, the combination of Herington with Romero does not present the present invention, in part for Herington is incapable of operating with or without Romero, as the present invention is so situated.

Accordingly, Applicant believes Herington, alone or in combination, does not disclose identifying a service level agreement negotiated with the client for the transaction application workload, the service level agreement specifying performance requirements for execution of the transaction application workload on the one server cluster.

V. ***Herington does not disclose assigning a subset of the server nodes in the one server cluster at the one domain to execute the transaction application workload;***

Incorporating the reasons above, Applicant respectfully disagrees with Examiner's position rejecting claim 34 in part that:

> "assigning a subset of the server nodes in the one server cluster at the one domain to execute the transaction application workload (load balancer monitors service levels provided by each node of cluster and routes transactions to one of the nodes based on the level of service that the node is providing, see, e.g., page 2, paragraph [0019]);"

Applicant notes that as is clearly understood by those skilled in the art, "assigning" and "monitoring" are not one and the same. In part, Applicant further points to the claims of the present invention themselves in which Applicant has patently distinguished the differences. For instance, from unamended claim 34:

> assigning a subset of the server nodes in the one server cluster at the one domain to execute the transaction application workload;

> monitoring execution of the transaction application workload on the subset of server nodes assigned to execute the transaction application workload to determine whether the performance requirements for execution of the transaction application workload specified in the service level agreement are being met; and

Applicant also believes Herington has equally set forth such commonly understood distinctions by stating:

[0025] "Partitions 510, 512 each include a workload manager 520, 522 to dynamically allocate resources ***assigned*** to the partition between their respective applications 530, 532 (on partition 510) and applications 534, 536 (on partition 512). "

[0029] In one embodiment, the method may further include monitoring service levels provided by each of the nodes 240, 242, 244, 246, 248 in the first cluster 220. Node 242 may then be selected based at least in part on its respective service level. Other performance metrics may also be used to select node 242 to satisfy the first transaction

Accordingly, Herington does not disclose assigning a subset of the server nodes in the one server cluster at the one domain to execute the transaction application workload.

VI. ***Herington does not disclose monitoring execution of the transaction application workload on the subset of server nodes assigned to execute the transaction application workload to determine whether the performance requirements for execution of the transaction application workload specified in the service level agreement are being met.***

Incorporating the reasons above, Applicant respectfully disagrees with Examiner's position rejecting claim 34 in part that:

"monitoring execution of the transaction application workload (performance information) on the subset of server nodes assigned to execute the transaction application workload to determine whether the performance requirements for execution of the transaction application workload specified in the service level agreement are being met (workload manger receives performance information from applications, see, e.g., page 2, paragraph [0023])"

Herington does not monitor the execution of the transaction application workload and Applicant further asserts that the assumption that such is "performance information" is mischaracterized. Herington provides for the "monitoring [of] service levels" at most at [0029]. Herington does not monitor the execution of applications, not the execution of applications which are transactional or parallel on one server cluster at one domain.

Herington also does not disclose teach or otherwise suggest "the subset of server nodes assigned to execute the transaction application workload to determine whether the performance requirements for execution of the transaction application workload specified in the service level agreement are being met." Examiner has stated that the "workload manger receives performance information from applications, see, e.g., page 2, paragraph [0023]" is equivalent as a basis for the rejection, but such foundation is misplaced. Applicant asserts that, in Herington, the workload manager merely receiving the performance information from application, as stated by Examiner, is not a determination as whether a performance requirement is met, is not a determination as to whether a performance execution has been met, and is not an act assigned to a subset of server nodes as in the present invention.

Further, Herington also does not disclose teach or otherwise suggest monitor the execution of the transaction application workload and Applicant further asserts that the assumption that such is "performance information" is mischaracterized. Herington provides for the "monitoring [of] service levels" at most at [0029]. Herington does not monitor the execution of applications, not the execution of applications which are transactional or parallel on one server cluster at one domain.

Accordingly, Herington does not disclose monitoring execution of the transaction application workload on the subset of server nodes assigned to execute the transaction application workload to determine whether the performance requirements for execution of the transaction application workload specified in the service level agreement are being met.

VII. ***Herington does not disclose provide for being responsive to a determination that the performance requirements for execution of the transaction application workload specified in the service level agreement are not being met, dynamically reassigning one or more of the server nodes in the one server cluster at the one domain assigned to execute the parallel application workload to the execution of the transaction application workload in order to meet the performance requirements for execution of the transaction application workload specified in the service level agreement.***

Incorporating the reasons above, Applicant respectfully disagrees with Examiner's position rejecting claim 34 in part that:

"responsive to a determination that the performance requirements for execution of the transaction application workload specified in the service level agreement are not being met, dynamically reassigning one or more of the server nodes in the one server cluster at the one domain assigned to execute the parallel application workload to the execution of the transaction application workload in order to meet the performance requirements for execution of the transaction application workload specified in the service level agreement (workload manager dynamically allocate and adjust computer resources between applications based on performance goals and performance information, see, e.g., page 2, paragraph [0024]);"

Applicant asserts that the allocation of system resources of Herington, is not performed in response to transaction application workloads and parallel application workloads, as set forth in the method of the present invention.

Accordingly, Herington does not disclose provide for being responsive to a determination that the performance requirements for execution of the transaction application workload specified in the service level agreement are not being met, dynamically reassigning one or more of the server nodes in the one server cluster at the one domain assigned to execute the parallel application workload to the execution of the transaction application workload in order to meet the performance requirements for execution of the transaction application workload specified in the service level agreement.

VIII. ***Herington does not disclose a method on one server cluster at one domain where wherein a server node assigned to execute the transaction application workload is not concurrently assigned to execute the parallel application workload and a server node assigned to execute the parallel application workload is not concurrently assigned to execute the transaction application workload.***

Incorporating the reasons above, Applicant respectfully disagrees with Examiner's position rejecting claim 34 in part that:

"wherein a server node (node 240 in figure 2) assigned to execute the transaction application workload (application 255 in figure 2) cannot be concurrently assigned to execute the parallel application workload (application 260 in figure 2) and a server node (node 250 in figure 2) assigned to execute the parallel application workload cannot be concurrently assigned to execute the transaction application workload (two node 240 and 250 are respectively dedicated to two different applications, see, e.g., page 1, paragraph [0014])."

Applicant has presented remarks to the cited reference of [0014] earlier, and again notes that Herington is unable to perform the method of the present invention on one server cluster at one domain, and that Herington does not disclose, teach or otherwise suggest such. Further, contrary to Examiner's assertions, the mere pictorialization of server configurations does not disclose the limitation of the present invention in which one server node of a transaction application workload is not concurrently assigned to execute a parallel application workload – in part, as Herington does not disclose, teach or suggest differing applications, and also as Herington interchangeably uses its similar if not identical applications in a different method than the present invention. Applicant has been unable to identify any teaching or reference as to such a suggestion in Herington.

Accordingly, Herington does not disclose a method on one server cluster at one domain where wherein a server node assigned to execute the transaction application workload is not concurrently assigned to execute the parallel application workload and a server node assigned to execute the parallel application workload is not concurrently assigned to execute the transaction application workload.

Additional Comments on Amended Claim 34

Applicant further incorporates the above remarks herein and further notes that neither Herington, nor Herington in combination with any of Examiner's cited art, disclose, teache or otherwise suggest the present invention, as none disclose:

"a method for supporting a transaction application workload and a parallel application workload on one server cluster including server nodes at one domain and configured to process the transaction application workload and the parallel application workload"

"a method comprising: receiving a request from a client to execute the transaction application workload on the one server cluster the transaction application comprising a plurality of discrete events that are less numerically intensive than the parallel application which comprises a numerically intensive application; identifying a service level agreement negotiated with the client for the transaction application workload, the service level agreement specifying performance requirements for execution of the transaction application workload on the one server cluster and acceptable performance of execution of the one server cluster in response to the transaction application and the parallel application; assigning a subset of the server nodes in the one server cluster at the one domain to execute the transaction application workload; monitoring execution of the transaction application workload on the subset of server nodes assigned to execute the transaction application workload to determine whether the performance requirements for execution of the transaction application workload and the acceptable performance of execution of the one server cluster in response to the transaction application and the parallel application specified in the service level agreement are being met; and responsive to a determination that the performance requirements for execution of the transaction application workload specified in the service level agreement are not being met, dynamically reassigning one or more of the server nodes in the one server cluster at the one domain assigned to execute the parallel application workload to the execution of the transaction application workload in order to meet the performance requirements for execution of the transaction application workload and the acceptable performance of execution of the one server cluster in response to the transaction application and the parallel application specified in the service level agreement, wherein a server node assigned to execute the transaction application workload cannot be concurrently assigned to execute the parallel application workload and a server node assigned to execute the parallel application workload cannot be concurrently assigned to execute the transaction application workload.

As Applicant believes claim 34 is allowable, as amended, as such in part further distinguishes the present invention to include aspects to the transaction application and the parallel application, Applicant further believes that all claims depending variously therefrom are also allowable for reasons similar.

CONCLUSION

On the basis of the above remarks, reconsideration and allowance of the claims is believed to be warranted and such action is respectfully requested. If the Examiner has any questions or comments, the Examiner is respectfully requested to contact the undersigned at the number listed below.

Applicant believes no fee to be due. The office is authorized to charge any underpayments and credit any overpayments to Deposit Account No. 09-0460.

Respectfully submitted,
SAWYER LAW GROUP LLP

Dated: July 31, 2008

/Joseph A. Sawyer, Jr./
Joseph A. Sawyer, Jr.
Attorney for Applicant
Reg. No. 30,801
(650) 475-1449